Random Bichromatic Matchings

Nayantara Bhatnagar[*,‡] Dana Randall[*] Vijay V. Vazirani[†] Eric Vigoda[‡]

April 27, 2006

Abstract

Given a graph with edges colored Red and Blue, we wish to sample and approximately count the number of perfect matchings with exactly k Red edges. We study a Markov chain on the space of all matchings of a graph that favors matchings with k Red edges. We show that it is rapidly mixing using non-traditional canonical paths that can backtrack. We also show that this chain can be used to sample dimer configurations on a 2-dimensional toroidal lattice with k Red edges, where the horizontal edges are Red and the vertical edges are Blue.

1 Introduction

Counting the number of matchings in a graph is a well-studied problem in combinatorics and computer science. Counting the number of perfect matchings in a bipartite graph is equivalent to computing the permanent of a matrix with $0, 1$ entries. This problem is also of interest in statistical physics in the context of understanding the thermodynamic properties of a dimer system [4, 5]. Motivated by this application, Kastelyn showed that for planar graphs, the number of perfect matchings can be computed exactly [11]. Recently Jerrum, Sinclair and Vigoda [8] gave an *fpras* (fully polynomial randomized approximation scheme) for approximating the number of perfect matchings in any bipartite graph, which is based on an *fpaus* (fully polynomial almost uniform sampler) for generating random perfect matchings.

A natural generalization of the matching problem arises when the edges of the graph are colored Red or Blue:

Problem: *Given a graph $G = (V, E)$, a partition $E = R \cup B$, and $k \leq \frac{|V|}{2}$, count the number of perfect matchings in G with exactly k edges in R.*

The decision version of this problem, known as *exact matchings*, is to find a matching with exactly k Red edges. This problem has been studied in combinatorial optimization [15] as well as statistical physics [3]. There are several open questions regarding both the decision and the counting versions of this problem. Mulmuley, Vazirani and Vazirani [14] give a randomized algorithm for exact matchings in general graphs. An intriguing open problem is whether their result can be derandomized. A deterministic algorithm for exact matchings is known only when the graph is complete or complete bipartite [10, 18].

A special case of the counting problem, of interest in statistical physics, is where G is the $\sqrt{n} \times \sqrt{n}$ 2-dimensional lattice and the horizontal edges are Red, while the vertical edges are Blue. We wish to count the number of dimer coverings with exactly k horizontal edges, as well as solve the sampling problem. Fisher [3] gave a closed form solution for the limiting (as the size of the lattice tends to infinity) *partition function* (see below) over configurations in terms of the activities λ and μ of horizontal and vertical dimers, where the weight of a configuration with k horizontal edges and k' vertical edges is given by $\lambda^k \mu^{k'}$.

We make progress on the counting problem for general graphs and solve the problem in some natural special cases. Our results for general graphs are best viewed in terms of the partition function for matchings. To state our results, we need some notation. Let $\mathcal{M}$ denote the set of all matchings of an input graph G,

[*]Supported in part by NSF grants CCR-0515105 and DMS-0505505. [†]Supported in part by NSF grants 0311541, 0220343 and CCR-0515186. [‡]Supported in part by NSF grant CCR-0455666.

and let $\mathcal{P}$ denote the set of perfect matchings. The standard partition function on matchings

$$Z_\lambda = \sum_{M \in \mathcal{M}} \lambda^{|M|}$$

can be approximated for all λ by the algorithm of Jerrum and Sinclair [7]. We show that we can approximate a modified partition function which puts most weight on matchings of size ℓ with exactly k RED edges, which we denote by *(k, ℓ)-matchings.*

Theorem 1.1 *For any $G = (V, E)$ with a partition of the edges $E = R \cup B$, activities $\lambda, \mu \le 1$, any $\ell \le |V|/2$ and $k \le \ell$, there is an fpras for estimating the following partition function over weighted matchings:*

$$Z_{\lambda,\mu}^{k,\ell}(\mathcal{M}) = \sum_{M \in \mathcal{M}} \lambda^{||M \cap R| - k|} \mu^{||M| - \ell|}. \tag{1}$$

For the problem of counting bichromatic perfect matchings, we show that in the special case of *dense* graphs we can estimate the partition function restricted to the set of perfect matchings. An n-vertex graph is dense if it has minimum degree $d_{min} \ge n/2$. A bipartite graph with each partition of size n is dense if it has $d_{min} \ge n/2$.

Theorem 1.2 *For any dense graph $G = (V, E)$, activity $\lambda \le 1$, and $k \le |V|/2$, there is an fpras for estimating the partition function*

$$Z_\lambda^k(\mathcal{P}) = \sum_{P \in \mathcal{P}} \lambda^{||P \cap R| - k|}. \tag{2}$$

We approximate the partition functions within a factor $(1 \pm \varepsilon)$ with probability at least $1 - \delta$. The running time in each case is polynomial in $1/\lambda, 1/\mu, 1/\varepsilon, \log(1/\delta)$ and the size of the graph.

We demonstrate the significance of these results on the 2-dimensional torus $\mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2}$ for even m_1, m_2, taking the horizontal edges to be RED and the vertical edges to be BLUE. In particular, we present a polynomial time algorithm for approximately sampling and counting the set of perfect matchings (or *dimer coverings*) with exactly k RED edges. We note that there are algorithms to exactly count the number of perfect matchings on the 2-d torus [11] which can be extended to bichromatic perfect matchings. However, our proof can be extended to the *monomer-dimer* model (see [7] for precise definitions and background) in which we approximately sample and count (k, ℓ)-matchings of the 2-d torus, giving the first solution to this more general problem.

Theorem 1.3 *Given any torus $\mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2}$ with m_1 and m_2 even, any non-negative integer $k \le m_1 m_2/2$ and any $\ell \ge k$, there is an fpaus for generating a random (k, ℓ)-matching of the torus and an fpras for estimating the number of such matchings that run in time polynomial in m_1 and m_2.*

We briefly describe the new technical contributions of our work. Theorem 1.1 uses a Markov chain defined on the set of all matchings of the graph which puts most weight on (k, ℓ)-matchings. We use the *canonical paths technique* to bound the convergence rate of the Markov chain. The crux of our analysis is the definition of the canonical paths, as opposed to the usual matching problem where analyzing the congestion along the paths is the harder task.

Let P and P' be two perfect matchings with exactly k edges. The traditional canonical path from P to P' defined in [7] starts at the matching P and alternately deletes an edge of P and adds an edge of P' along an alternating cycle. This ensures that we stay in the set of perfect or near perfect matchings. However, there are no guarantees about the number of RED edges in the intermediate matchings along the path. We need a way of defining such a path that only uses matchings that are close to perfect while at the same time maintaining the number of RED edges close to k. The combinatorial fact that enables us to define our paths is as follows: For any k and for all perfect matchings P, P' with exactly k RED edges, there is a polynomial length path between P and P' along almost perfect matchings, with successive matchings differing by only a few edges, such that each contains close to k RED edges. We reduce the problem of finding such a path to a combinatorial problem about moving two points along a two-dimensional landscape in a co-ordinated manner so that the sum of their heights stays constant. An interesting aspect of the resulting canonical paths is that they may backtrack along portions of the alternating cycle, for instance we might delete edges of P' that were previously added.

Our second technical contribution is proving combinatorial inequalities that allow us to approximate the number of (k,ℓ)-matchings on the torus. Kenyon, Randall and Sinclair [12] showed that the number of near perfect matchings in the d-dimensional torus is polynomially related to the number of perfect matchings, thereby yielding polynomial time algorithms for approximately counting and uniformly sampling perfect matchings. In this paper, we generalize their result to show that, on the 2-d torus, this relationship holds even when we restrict to sets of matchings with exactly k RED edges. Our result builds on ideas of Temperley [17] and Burton and Pemantle [2] for constructing augmenting paths where every horizontal and vertical segment has even length.

2 Markov Chains

The basis of the sampling and counting algorithms are Markov chains on sets of bichromatic matchings. We first introduce some basic definitions and results about Markov chains.

The *total variation distance* between two distributions μ, ν on Ω is given by

$$d_{tv}(\mu,\nu) = \frac{1}{2}\sum_{x\in\Omega}|\mu(x)-\nu(x)|$$

Let P denote the transition matrix of the chain MC, and thus $P^t(x,\cdot)$ denotes the distribution after t steps of the chain, with starting state x. The *mixing time* $\tau_x(\delta)$ of MC starting at state $x\in\Omega$ is defined as

$$\tau_x(\delta) = \min\{t\geq 0 \mid d_{tv}(P^t(x,\cdot),\pi)\leq\delta\}$$

In the canonical path method for bounding the mixing time of a Markov chain, for each pair of matchings I and F in the state space Ω, we define a path from I to F along transitions of the chain. We then bound the *congestion* of these paths through every transition to show that no edge is overloaded. This is enough to bound the convergence rate of the Markov chain.

Let $\gamma_{I,F}$ denote this canonical path from I to F and let Γ denote the set of paths. Denote the set of pairs of matchings whose canonical paths use the transition t by

$$cp(t) = \{(I,F)\in\Omega\times\Omega : \gamma_{I,F} \text{ uses the transition } t\}.$$

Define the *congestion* of Γ to be

$$\rho(\Gamma) := \max_t\left\{\frac{1}{\pi(M)P(M,M')}\sum_{(I,F)\in cp(t)}|\gamma_{I,F}|\pi(I)\pi(F)\right\}. \tag{3}$$

This implies the following bound on the mixing time, see for example [16],

$$\tau_x(\delta)\leq\rho(\Gamma)(\log\pi(x)^{-1}+\log\delta^{-1})$$

To sample from the distribution on matchings given by Equation (1), we define a Markov chain $\mathcal{T}_1$ on the set of matchings $\mathcal{M}$ which puts the most weight on the (k,ℓ)-matchings. A related chain was studied by Jerrum and Sinclair [7] with the transition probabilities defined so that the stationary distribution was uniform over all matchings. Let $\mathcal{N}(u,v)$ denote the set of matchings of all sizes in which the vertices u, v are unmatched.

The Markov Chain $\mathcal{T}_1$: The state space is $\mathcal{M}$, the set of all matchings of G. Let $\ell\leq|V|/2$, $0\leq k\leq\ell$ and $0<\lambda,\mu\leq 1$. Define the weight of a matching M, as

$$w(M) = w_{\lambda,\mu}(M) = \lambda^{||M\cap R|-k|}\mu^{||M|-\ell|}.$$

The transitions $M_t\rightarrow M_{t+1}$ of $\mathcal{T}_1$ are defined as follows.

Suppose we are at the matching M_t. Choose a random edge $e=(u,v)\in E$ of the graph G.

1. Delete an edge: If $e \in M_t$ set $M' = M_t \setminus \{e\}$.
2. Add an edge: If u and v are unmatched, set $M' = M_t \cup \{e\}$.
3. Switch two edges: If $z \neq v$, $M_t \in \mathcal{N}(u,z)$ and $(w,v) \in M_t$, set $M' = (M_t \cup \{e\}) \setminus (w,v)$.
4. Set $M_{t+1} = M'$ with probability $\frac{1}{2}\min(1, w(M')/w(M))$, else set $M_{t+1} = M_t$.

It is straightforward to verify that the Markov chain is connected, aperiodic and reversible and hence has stationary distribution proportional to $w(M)$. We prove the following theorem:

Theorem 2.1 *For any $G = (V,E)$ with a partition of the edges $E = R \cup B$, activities $\lambda, \mu \leq 1$, any $\ell \leq |V|/2$ and $k \leq \ell$, the Markov chain $\mathcal{T}_1$ mixes in time polynomial in the size of the graph, $1/\lambda$ and $1/\mu$.*

To sample from the distribution on matchings given by Equation (2), we define a new Markov chain $\mathcal{T}_2$. Let $\mathcal{N}^i$ be the set of matchings with $2i$ unmatched vertices.

The Markov Chain $\mathcal{T}_2$: The state space is $\cup_{i=1}^{4} \mathcal{N}^i \cup \mathcal{P}$, i.e. the set of matchings of G with at most eight unmatched vertices (we assume the underlying graph has an even number of vertices, so the number of unmatched vertices must be even). Let $0 \leq k \leq |V|/2$ and $0 < \lambda \leq 1$. Define the weight of a matching M, as

$$w(M) = w_\lambda(M) = \lambda^{||M \cap R| - k|}.$$

The transitions $M_t \to M_{t+1}$ of $\mathcal{T}_2$ are defined as follows.

Suppose we are at the matching M_t. Choose a random edge $e = (u,v) \in E$ of the graph G.

1. Delete an edge: If $e \in M_t$ set $M' = M_t \setminus \{e\}$ provided M' has at most eight unmatched vertices.
2. Add an edge: If u and v are unmatched, set $M' = M_t \cup \{e\}$.
3. Switch two edges: If $z \neq v$, $M_t \in \mathcal{N}(u,z)$ and $(w,v) \in M_t$, set $M' = (M_t \cup \{e\}) \setminus (w,v)$.
4. Set $M_{t+1} = M'$ with probability $\frac{1}{2}\min(1, w(M')/w(M))$, else set $M_{t+1} = M_t$.

This chain is also reversible and aperiodic. We will later show that it is connected. Hence it has stationary distribution proportional to $w(M)$. It turns out that to define canonical paths between perfect matchings and bound the congestion, we may need intermediate matchings on the path with up to 8 unmatched vertices, and this is the reason the state space of $\mathcal{T}_2$ includes these matchings. We prove the following theorem:

Theorem 2.2 *For any dense graph $G = (V,E)$, activity $\lambda \leq 1$, and $k \leq |V|/2$, the Markov chain $\mathcal{T}_2$ mixes in time polynomial in the size of the graph and $1/\lambda$.*

2.1 Reducing Counting to Sampling

By standard reductions, approximating the partition functions $Z^{k,\ell}_{\lambda,\mu}(\mathcal{M})$ and $Z^{k}_{\lambda}(\mathcal{P})$ can be reduced to approximate sampling, see for instance [6]. We sketch the reductions, and then focus on the sampling problems for the remainder of the paper.

Suppose our goal is to estimate $Z^{k,\ell}_{\lambda,\mu}(\mathcal{M})$. Let $1 = \lambda_1 > \cdots > \lambda_t = \lambda$ and $1 = \mu_1 > \cdots > \mu_{t'} = \mu$. We can write $Z^{k,\ell}_{\lambda,\mu}(\mathcal{M})$ as a telescoping product for varying values of λ_i as follows:

$$Z^{k,\ell}_{\lambda,\mu}(\mathcal{M}) = \frac{Z^{k,\ell}_{\lambda_t,\mu_{t'}}(\mathcal{M})}{Z^{k,\ell}_{\lambda_{t-1},\mu_{t'}}(\mathcal{M})} \times \cdots \times \frac{Z^{k,\ell}_{\lambda_2,\mu_{t'}}(\mathcal{M})}{Z^{k,\ell}_{\lambda_1,\mu_{t'}}(\mathcal{M})} \times Z^{k,\ell}_{\lambda_1,\mu_{t'}}(\mathcal{M}). \tag{4}$$

Further we can write the last term $Z^{k,\ell}_{\lambda_1,\mu_{t'}}(\mathcal{M})$ above as a telescoping product over varying values of μ_i:

$$Z^{k,\ell}_{\lambda_1,\mu_{t'}}(\mathcal{M}) = \frac{Z^{k,\ell}_{\lambda_1,\mu_{t'}}(\mathcal{M})}{Z^{k,\ell}_{\lambda_1,\mu_{t'-1}}(\mathcal{M})} \times \cdots \times \frac{Z^{k,\ell}_{\lambda_1,\mu_2}(\mathcal{M})}{Z^{k,\ell}_{\lambda_1,\mu_1}(\mathcal{M})} \times Z^{k,\ell}_{\lambda_1,\mu_1}(\mathcal{M}). \tag{5}$$

The term $Z^{k,\ell}_{\lambda_1,\mu_1}(\mathcal{M})$ is simply the total number of matchings in the graph which can be estimated using the algorithm of Jerrum and Sinclair [7]. We set

$$\lambda_i = \mu_i = \left(1 - \frac{1}{3n}\right)^i.$$

This implies that

$$\frac{1}{2} \le \frac{w_{\lambda_i,\mu_j}(M)}{w_{\lambda_{i-1},\mu_j}(M)} \le 1.$$

Then, by Chebyshev's inequality, by sampling from the set $\mathcal{M}$ with the chain $\mathcal{T}_1$ at λ_i, u_j we can estimate the ratio $Z^{k,\ell}_{\lambda_i,\mu_j}(\mathcal{M})/Z^{k,\ell}_{\lambda_{i-1},\mu_j}(\mathcal{M})$ to within a factor $1 \pm \varepsilon/3t$ with $O(t\varepsilon^{-2})$ samples. (See [6, 8] for details of this standard calculation.) A similar inequality is implied for $w_{\lambda_i,\mu_j}(M)/w_{\lambda_i,\mu_j}(M)$, allowing us to approximate the ratios $Z^{k,\ell}_{\lambda_i,\mu_j}(\mathcal{M})/Z^{k,\ell}_{\lambda_i,\mu_{j-1}}(\mathcal{M})$. Plugging these estimates into Equations (4) and (5) gives a $1 \pm \varepsilon$ approximation to $Z^{k,\ell}_{\lambda,\mu}(\mathcal{M})$.

To estimate $Z^k_\lambda(\mathcal{P})$, let $1 = \lambda_1 > \cdots > \lambda_t = \lambda$. Let Ω denote the state space of the chain $\mathcal{T}_2$. Then we have the following telescoping product:

$$Z^k_\lambda(\Omega) = \frac{Z^k_{\lambda_t}(\Omega)}{Z^k_{\lambda_{t-1}}(\Omega)} \times \cdots \times \frac{Z^k_{\lambda_2}(\Omega)}{Z^k_{\lambda_1}(\Omega)} \times Z^k_{\lambda_1}(\Omega). \tag{6}$$

The term $Z^k_{\lambda_1}(\Omega)$ is the number of matchings in the graph with at most 8 unmatched vertices. Since the graph is dense, this can be estimated using the matching algorithm of Jerrum and Sinclair [7]. Moreover, since the graph is dense, the ratio $Z^k_\lambda(\mathcal{P})/Z^k_\lambda(\Omega)$ is at least inverse polynomial, and hence it can be estimated by sampling. Thus, we can obtain an approximation to $Z^k_\lambda(\mathcal{P})$, as desired.

3 The Canonical Paths

Jerrum and Sinclair define canonical paths on the set of all matchings in order to analyze their Markov chain in the monochromatic setting [7]. However, if we were to use these paths, there are no guarantees regarding the number of Red edges in the intermediate matchings. We need such a guarantee in order to bound the congestion. In this section, we consider a special case which illustrates why we need different canonical paths and gives the idea behind the general construction. We will focus on defining these paths for $\mathcal{T}_1$. The paths for $\mathcal{T}_2$ involve a further small modification and will be defined later.

3.0.1 Intuition for the Canonical Paths

The approach of Jerrum and Sinclair [7] to obtain a bound on the congestion is to focus on a specific transition T. For each pair (I, F) whose path uses the transition T, we define an "encoding" E, which is also a matching; the configurations T and E let us recover I and F, so E can be viewed as an injective map. Then the number of (I, F) pairs whose path uses T is at most the number of matchings, which is $|\Omega|$. This is sufficient to bound the congestion for unweighted matchings. For weighted matchings, we need to show that

$$w(I)w(F) \le w(T)w(E)poly(n).$$

The encoding is defined as $E = (I \cup F) \setminus (M \cup M')$, where T is the transition from M to M'. Therefore E can be viewed as a matching which is the complement of T, or more precisely $M \cup M'$ with respect to $I \cup F$.

Suppose that $\ell = |V|/2$ so that the partition function favors perfect matchings. If I and F are perfect matchings with k Red edges, they have maximum weight. For the above inequality to be satisfied, the weight of transitions and encodings along the canonical path from I to F must be comparable to the weight of I and F. Hence, both M and E need to contain close to k Red edges, and simultaneously be close to a perfect matching (i.e., they have only a constant number of unmatched vertices).

Consider the perfect matchings I and F, and suppose $I \oplus F$ (the symmetric difference of I and F) consists of a single alternating cycle. The transitions of the chain allow us to easily "unwind" this alternating cycle:

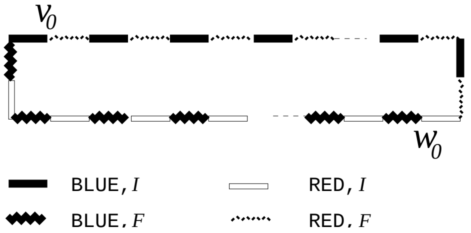

Figure 1: An alternating cycle in $I \oplus F$

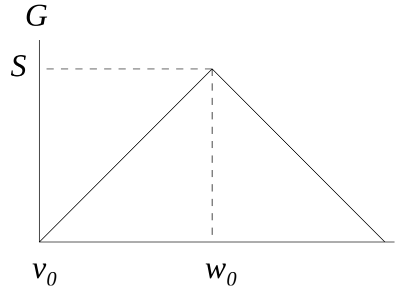

Figure 2: Landscape for cycle

remove one of the edges of I on the cycle, then perform a series of shifts (moves of type 3), and then add the final edge of the cycle of F.

Suppose, as in Figure 1, this cycle alternates RED on I and BLUE on F on one half of the cycle, and BLUE on I and RED on F on the other. Then, one can verify that no matter where we start the unwinding there will be some intermediate matching with far more (or far less) RED edges than the intended k. However, note that in this example there are *two* vertices v_0, w_0 so that if we unwind from these two points *simultaneously* then we can ensure that the number of RED edges differs from k by at most a constant. It turns out that we can always choose two such positions to begin the unwinding of the cycle. This idea forms the basis of our solution. However, the protocol for unwinding is not straightforward and we may need need to backtrack (switch edges back from F to I) from one position to continue unwinding at the other. Hence, it is not obvious whether our paths can always make progress. Additionally, the picture is more complicated when $I \oplus F$ consists of multiple cycles and paths with varying lengths and numbers of RED edges. We focus on first formalizing the problem of unwinding a single alternating cycle and then provide the general solution.

Suppose $I \oplus F$ is a single alternating cycle and I and F both contain exactly k RED edges. We would like to transform the cycle from I to F so that all the intermediate matchings have close to k RED edges. To define the unwinding, it is helpful to look at the alternating cycle together with a function representing the number of RED edges gained along the cycle as we replace edges. Fix a direction to unwind in, and for every alternate vertex v on the alternating cycle, assign $-1, 0$ or $+1$ to denote the change in the number of RED edges when one of the edges incident with it is added and the other deleted. Thus, for $e = (u, v) \in I, e' = (v, w) \in F$, $f(v) = 1_{e' \in R} - 1_{e \in R}$, where 1 is the indicator function. Fix a start vertex on the cycle, say v_0, and a direction for unwinding the cycle. For every vertex $v_{2\ell+1}$ on the cycle, let $G(v_{2\ell+1}) = \sum_{i=0}^{\ell} f(v_{2i+1})$, where $v_0 \to v_1 \to \cdots \to v_\ell$ is the alternating path from v_0 to v_ℓ. The function G defines a "landscape", as shown in Figure 2.

It is an easy exercise to show that if $|I \cap R| = |F \cap R| = k$, then there always exists a vertex v_0 so that $G(v_0) = 0$, $G(v_\ell) \geq 0$ for all ℓ, and 0 again at the last vertex. We choose a companion start vertex for v_0 which is a (global) maximum, denote this vertex as w_0. Let $S = G(v_0) + G(w_0)$. We break the alternating cycle into a pair of alternating paths, $P = \{v_0, v_1, \ldots, v_n\}$ and $Q = \{w_0, \ldots, w_m\}$, where v_n is the vertex before w_0 and w_m is the vertex before v_0.

We now start unwinding the cycle at the vertices v_0 and w_0. If unwinding from one of the positions adds a RED edge, then from the other position we need to remove a RED edge by moving forward or backwards as necessary. Thus, if at some intermediate step we are at vertices v_i and w_j, we need that $(G(v_i) - G(v_0)) + (G(w_j) - G(w_0)) = 0$, i.e. $G(v_i) + G(w_j) = S$. The mountain climbing problem is to determine a (short) trajectory from (v_0, w_0) to (v_n, w_m) so that at each intermediate step (v_i, w_j) we have $G(v_i) + G(w_j) = S$. We may need to move backwards on one path in order to move forward on the other path, and this corresponds to "rewinding" the cycle. In the next section we prove that such a trajectory can in fact always be found.

3.0.2 The Algorithm for Mountain Climbing

A *landscape* is a function $P : [n] \to \mathbb{Z}_{\geq 0}$ such that for $1 \leq i \leq n-1$, $|P(i+1) - P(i)| = 1$ (see Figure 3). For $n, m > 1$, given landscapes $P : [n] \to \mathbb{Z}_{\geq 0}$ and $Q : [m] \to \mathbb{Z}_{\geq 0}$, we say P and Q are *S-matched* if there is an integer S s.t.

i) $P(1) + Q(1) = P(n) + Q(m) = S$

ii) $P(1) = \min_i\{P(i)\}$, $P(n) = \max_i\{P(i)\}$, $Q(1) = \max_j\{Q(j)\}$, $Q(m) = \min_j\{Q(j)\}$.

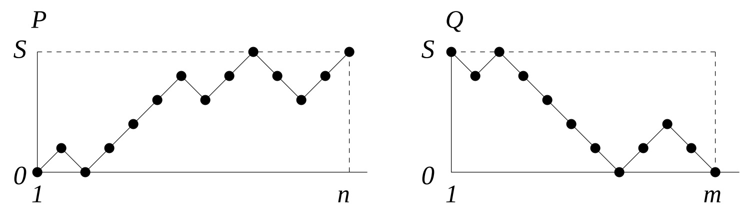

Figure 3: S-matched landscapes P, Q

A *traversal* of S-matched landscapes P, Q is a sequence $(i_1, j_1), \cdots, (i_\ell, j_\ell)$, s.t.

i) $i_1 = 1$, $j_1 = 1$, $i_\ell = n$, $j_\ell = m$

ii) For $1 \leq k \leq \ell - 1$, $|i_{k+1} - i_k| = 1$, $|j_{k+1} - j_k| = 1$ and $P(i_k) + Q(j_k) = S$.

Lemma 3.1 *Let P and Q be S-matched landscapes on $[n]$ and $[m]$ respectively. Then, there exists a traversal of P and Q of length at most $O(nm)$ and it can be found in time $O(nm)$.*

Our original proof was more complicated [1]. The simpler proof here is due to Martin Matamala [13].

Proof: We translate the lemma to a connectivity problem in a graph on pairs of points. Consider the graph whose vertices are the pairs of points $(i, j) \in [n] \times [m]$ such that $P(i) + Q(j) = S$. The vertex (i_1, j_1) is adjacent to (i_2, j_2) if $i_1 = i_2 \pm 1$ and $j_1 = j_2 \pm 1$. Then, since P and Q are S-matched, the vertices $(1, 1)$ and (n, m) are each of degree 1. Every other vertex is of degree 0, 2 or 4. This is certainly clear for all vertices (i, j) where $i \in [n] \setminus \{1, n\}, j \in [m] \setminus \{1, m\}$. For the other vertices, where one coordinate is an endpoint, and the other is not, the degree is 2 since the endpoints are either minimums or maximums, while the coordinate which is not an endpoint must be a "peak" or a "valley."

Since there are only 2 odd degree vertices, they must be in the same component. Thus there is a path from $(1, 1)$ to (n, m), which is exactly the required traversal. The path can be found in time linear in the possible number of edges, and thus in $O(nm)$ time. ∎

We can now define the canonical path between two perfect matchings I, F when both have k RED edges and $I \oplus F$ is a single alternating cycle. It is simply the sequence of additions and deletions along the alternating cycle dictated by the algorithm for mountain climbing, where we open up the cycle at the vertices v_0 and w_0 defined previously. It is clear that since the sum $P(i_t) + Q(j_t)$ remains constant along the traversal, the number of RED edges in the intermediate matchings will differ from k by at most a small constant.

4 Defining Canonical Paths Between Arbitrary Matchings

The solution to the mountain climbing problem has all the ingredients we need to define the canonical paths for general matchings I, F. The canonical paths are defined so that every pair of successive matchings along the path is a transition of the Markov chain and the size of an intermediate matching lies between the sizes of I and F and consists of $[I_R - 5, F_R + 1]$ RED edges, where $I_R = |I \cap R|, F_R = |F \cap R|$ and $I_R \leq F_R$. Essentially, we think of the concatenation of all the paths and cycles of $I \oplus F$ as one long landscape, and apply Lemma 3.1 without ever unwinding more than constantly many cycles or paths at any time. By the previous argument, the canonical paths are at most of polynomial length. With standard machinery it is then straightforward to show that the Markov chain mixes in polynomial time.

Consider $I \oplus F$, which is a set of alternating cycles and paths, $C_1, \cdots, C_z$. We want to unwind the cycles, shifting edges from I to F. To define the unwinding, we will first define the auxiliary "labeled" paths $D_1, \cdots, D_z$. The paths D_i will enable us to reformulate the problem of unwinding the cycles and paths in a form convenient for applying the algorithm for mountain climbing. Let $C_i = v_1^i, \cdots, v_{2t_i}^i$ or $v_1^i, \cdots, v_{2t_i+1}^i$ if it is a cycle or a path whose first edge is in I. If C_i is a path whose first edge is in F, number it either $v_2^i, \cdots, v_{2t_i}^i$ or $v_2^i, \cdots, v_{2t_i+1}^i$ depending on whether it is odd or even length. Define the path D_i with vertices[1] $w_1^i, \cdots, w_{t_i}^i$. Each edge (w_j, w_{j+1}) of D_i is labeled ℓ_j^i for $1 \leq j < t$, where ℓ_j^i denotes the change in the number of RED edges when the edge (v_{2j}, v_{2j+1}) is added and (v_{2j+1}, v_{2j+2}) is deleted. More formally,

$$\ell_j^i = \begin{cases} 1 & \text{if } (v_{2j+1}, v_{2j+2}) \in B \text{ and } (v_{2j}, v_{2j+1}) \in R; \\ -1 & \text{if } (v_{2j+1}, v_{2j+2}) \in R \text{ and } (v_{2j}, v_{2j+1}) \in B; \\ 0 & \text{if both } (v_{2j+1}, v_{2j+2}), (v_{2j}, v_{2j+1}) \in R \text{ or both are in } B. \end{cases}$$

In addition, we define two extra labels $\ell_0^i, \ell_{t_i}^i$, not associated with any edge of D_i which are defined to account for the change in the number of RED edges at the beginning or end of a path or a cycle when a single edge is added or deleted.

1. Suppose that C_i is a cycle. Let $C_i = v_1, \cdots, v_{2t_i}$ so that $(v_{2k-1}, v_{2k}) \in I$ for $1 \leq k \leq t$ and $(v_{2k}, v_{2k+1}) \in F$ for $1 \leq k \leq t$. Let

$$\ell_0^i = \begin{cases} -1 & \text{if } (v_1, v_2) \in R; \\ 0 & \text{if } (v_1, v_2) \in B. \end{cases}$$

$$\ell_{t_i}^i = \begin{cases} 1 & \text{if } (v_{2t_i}, v_1) \in R; \\ 0 & \text{if } (v_{2t_i}, v_1) \in B. \end{cases}$$

2. Suppose C_i is a path. We define the extra labels $\ell_0^i, \ell_{t_i}^i$ depending on whether the path is even or odd and whether the first edge is to be deleted or added.
   (a) If C_i has an even number of vertices,
      i) If the first edge of $C_i = v_1, \cdots, v_{2t_i}$, (v_1, v_2) is in I, let $\ell_{t_i}^i = 0$ and let $\ell_0^i = -1$ if $(v_1, v_2) \in R$ and 0 if $(v_1, v_2) \in B$.
      ii) If the first edge of $C_i = v_2, \cdots, v_{2t_i+1}$, (v_2, v_3) is in F, let $\ell_0^i = 0$ and let $\ell_{t_i}^i = 1$ if $(v_{2t_i}, v_{2t_i+1}) \in R$ and 0 if $(v_{2t_i}, v_{2t_i+1}) \in B$.
   (b) If C_i has an odd number of vertices,
      i) If the first edge of $C_i = v_1, \cdots, v_{2t_i+1}$, (v_1, v_2) is in I, let

$$\ell_0^i = \begin{cases} -1 & \text{if } (v_1, v_2) \in R; \\ 0 & \text{if } (v_1, v_2) \in B. \end{cases}$$

$$\ell_{t_i}^i = \begin{cases} 1 & \text{if } (v_{2t_i}, v_{2t_i+1}) \in R; \\ 0 & \text{if } (v_{2t_i}, v_{2t_i+1}) \in B. \end{cases}$$

[1] Throughout, we omit the superscript i, and the subscript on t if the cycle is clear from context.

ii) If the first edge of $C_i = v_2, \cdots, v_{2t_i}$, (v_2, v_3) is in F, let $\ell_0^i = \ell_{t_i}^i = 0$.

We want to think of unwinding a cycle or path C_i as a token moving around on the vertices of D_i. For example, we would like that moving the token from w_j to w_{j+1} in D corresponds to the move in the Markov chain which switches the edge (v_{2j+1}, v_{2j+2}) in C to the edge (v_{2j+1}, v_{2j}) (or, alternatively, the unmatched vertex at v_{2j} moves to $v_{2(j+1)}$). The label ℓ_j indicates how the current number of RED edges changes when such a move is made during the unwinding of the path. Completely unwinding C_i, i.e. going from I to F corresponds to the token visiting every vertex in D_i. In addition, in contrast to the usual canonical paths, we would like to stay (close to) in between the number of RED edges in the initial and final matchings. To achieve this, instead of unwinding the cycles and paths in order as usual, we will use one or more markers, or tokens, to unwind more than one cycle or path at a time. We divide all the cycles and paths into segments which will be unwound disjointly. We use the tokens to mark the beginning of the segment which is currently being unwound and the position to which the unwinding has progressed. It may be necessary to reverse the unwinding, i.e. wind back a portion by sliding edges from F to I on one cycle, while making progress in unwinding another cycle.

In order to bound the congestion, we will need to ensure that the size of intermediate matchings on the canonical path are bounded. To do this, we impose the following condition on the order of cycles and paths in $I \oplus F$. Suppose that $|I| \leq |F|$ (the other case can be defined analogously). We order first all the alternating cycles and alternating paths with an even number of edges. Since $|I| \leq |F|$ there are exactly $|F| - |I|$ more odd length alternating paths with an extra edge of F than paths with an extra edge of I. Next in the order are (canonically defined) $|F| - |I|$ odd length alternating paths each with an extra edge of F. Lastly, we order the remaining odd length alternating paths so that they alternately contain an extra edge of I or an extra edge of F. When we define the canonical paths, we will see that this ordering ensures that the size of intermediate matchings is bounded.

Let P be the path obtained by concatenating the D_i in the above canonical order, and adding an extra initial vertex w_0 and an extra final vertex w_T. Let the vertices of P be $w_0, w_1, \cdots, w_T$, and let ℓ_i be the label on the edge (w_i, w_{i+1}) for $0 \leq i < T$. The edge joining $w_{t_i}^i$ from D_i and w_1^{i+1} from $D_{i'}$ has label $\ell_{t_i}^i + \ell_0^{i'}$. Let $\ell_0 = \ell_0^1$ and $\ell_{T-1} = \ell_{t_z}^z$. Every other edge of P takes its label from the path it is in. Note that $\ell_i \in \{-1, 0, 1\}$. With each vertex w_i of P for $1 \leq i \leq T$, associate a value $f(w_i) = f(w_{i-1}) + \ell_{i-1}$, letting $f(w_0) = 0$. Note that $f(w_T) = F_R - I_R$.

We will visit the vertices of P in rounds, using 4 tokens in every round i; two "start" tokens, A_{2i-1}, A_{2i} and two "moving" tokens B_{2i-1}, B_{2i}. The start tokens mark the beginning of the segment being unwound while the moving tokens will mark the position until which the unwinding has progressed. To define the rounds, divide the path P into a union of edge disjoint paths[2] $Q_i = a_1^i, \cdots, a_{m_i}^i$ such that for every $a_j^i = w_{j'}$, $f(w_{j'}) \geq 0$ or $f(w_{j'}) \leq 0$. Note that $a_{m_i}^i = a_1^{i+1}$ for all i except the last. In the i-th round, we use the 4 tokens to visit the vertices of Q_i. We start by describing the movement of the tokens in the first round and then assuming we have computed the i^{th} round, we define how the tokens move in the $(i+1)^{st}$ round.

In the 1st round, A_1, B_1 are placed at $w_0 = a_1^1$, the first vertex of Q_1. The tokens A_2, B_2 are placed at some other vertex of Q_1, which we can choose. The tokens A_1, A_2 stay fixed, while B_1, B_2 are moved (forwards or backwards) one edge and one token at a time, so that the sum of the labels they encounter is in the range $[I_R - 4, F_R + 1]$ (if B_1, B_2 traverse an edge in the backward direction, the sign of the label is multiplied by -1). B_1, B_2 never move to a position before the corresponding start tokens A_1, A_2. B_1 does not move beyond the position of A_2, and B_2 does not move beyond the last vertex of Q_1, $a_{m_1}^1$. We say that Q_1 has been covered if at some time when B_2 is at $a_{m_1}^1$, we can move B_1 to the position of A_2, or at some time when B_1 is at the position of A_2, we can move B_2 to $a_{m_1}^1$. The round ends after this step is performed.

Suppose at the end of the i-th round, A_{2i-1} is at a_1^i, B_{2i-1}, A_{2i} are at the same position and B_{2i} is at $a_{m_i}^i$. Start the $i+1$-th round with A_{2i+1}, B_{2i+1} at $a_1^{i+1} = a_{m_i}^i$, and A_{2i+2}, B_{2i+2} at some other vertex of Q_{i+1}. In the $i+1$-th round, the tokens are allowed to move in the same fashion, i.e. the tokens A_{2i+1}, A_{2i+2} stay fixed, while the tokens B_{2i+1}, B_{2i+2} move forwards or backwards one at a time, B_{2i+1}, B_{2i+2} never move to a position before A_{2i+1}, A_{2i+2} respectively, B_{2i+2} does not move beyond the position of A_{2i+2} and B_{2i-1} does not move beyond the last vertex of Q_i, $a_{m_{i+1}}^{i+1}$. The objective is to cover the path Q_{i+1}.

[2] There may be more than one way to divide, choose one arbitrarily

We would like to show that to show there is a sequence of moves for the tokens so that the path P can be covered, by covering each path Q_i. Finally, we will show that unwinding the cycles and paths of $I \oplus F$ can be reduced to the problem of covering the path P in this fashion. Lemma 3.1 will be used to show that we can cover the paths Q_i using the tokens B_q.

We wish to use the lemma to cover all the paths Q_i defined previously, so that the movements of the tokens B_{2i-1}, B_{2i} are a traversal of two S-matched landscapes that we will specify. Lemma 3.1 assumes edges have labels ± 1, so we assume that a token always moves across as many edges as it can in the same direction until it encounters a non-zero label. Also, when making a move, the nearest non-zero label is taken into account (forwards or backwards). Let $Q_i = a_1, \cdots, a_m$ be the path to be covered. There are two cases: in the first, P is a path corresponding to no change in the number of RED edges. In the second case, P is a path where there is a net gain or loss in the number of RED edges.

1. Consider the case that $f(w_T) = 0$. Let $a_{\max}$ be the last vertex of Q_i where $|f(w_j)|$ is maximized for $1 \leq j \leq m$. If $a_{\max} = a_m$, all the labels are 0. In this case, simply use 1 token, B_{2i-1} to go from a_1 to a_m while A_{2i}, B_{2i} are placed at a_m and B_{2i} does not move. Otherwise, start with the tokens A_{2i-1}, B_{2i-1} at a_1 and the tokens A_{2i}, B_{2i} at $a_{\max}$. Then, Q_i can be covered as in Lemma 3.1 where the functions P, Q of the Lemma are the function f on the segments $a_1, \cdots, a_{\max}$ and $a_{\max}, \cdots, a_m$ respectively, with $S = f_{\max}$.

2. Now assume that $f(w_T) = F_R - I_R > 0$. Again, we decompose P into a union of paths Q_i. However, the last path, Q_k no longer satisfies the condition of the Lemma that $f(w_T) = 0$. If every $f(w_j) \leq f(w_T)$, we simply move one token, B_{2k-1} from a_1 to a_T, place A_{2k}, B_{2k} at a_T and never move B_{2k}. Otherwise, let $a_{\max}$ be the last vertex where $|f(w_j)|$ is maximized for $1 \leq j \leq m$. Note that $f(a_{\max}) > f(w_m)$. Start with A_{2k-1}, B_{2k-1} at a_1 and A_{2k}, B_{2k} at $a_{\max}$. We now apply Lemma 3.1 to the portions of the path $a_1, \cdots, a_{\max}$ which have value $f \leq f(a_{\max}) - (F_R - I_R)$ and the subpath $a_{\max}, \cdots, a_m$. For the portions of the path $a_1, \cdots, a_{\max}$ whose "height" is greater than $f(a_{\max}) - (F_R - I_R)$ (and less that or equal to $f(a_{\max})$), we only move the token B_{2i-1}, keeping B_{2i} fixed. Thus for every Q_i except possibly the last, the sum of the labels traversed by the tokens B_{2i-1}, B_{2i} is 0. For the last path, the sum is no more than $F_R - I_R$.

Next, we define how the moves of the tokens on P correspond to moves of the Markov chain $\mathcal{T}_1$. We will refer to the paths C_k as odd or even, depending on whether they have an odd or even number of *vertices*. Let the "path", $M_k = C_k \cap I$ for $1 \leq k \leq z$. A *hole* in M_k is a vertex which is matched in I but unmatched in M_k. In each step of the unwinding, when a token moves forward, edges in F will be added, and edges of I removed. When a token moves backwards, edges of I are possibly added and edges of F removed.

---

1. When two new tokens A_{2i}, B_{2i} are placed at w_j, a vertex in D_k, remove the edge (v_{2j-1}, v_{2j}) from M_k if it is present.

2. B_{2i-1} or B_{2i} moves forward or backward in the i-th round.

   a) If the token moves forward from w_j to w_{j+1} in D_k,

      i) Remove the edge (v_{2j+1}, v_{2j+2}) from M_k if it is present. It may be absent if A_{1-q}, B_{1-q} were placed at w_{j+1}. Note that in that case there is a hole at v_{2j+1}.

      ii) Add (v_{2j}, v_{2j+1}) to M_k. Note that by the previous step, there is a hole at v_{2j+1}.

   b) If the token moves forward from $w^k_{t_k}$ in D_k (or from w_0) to $w^{k'}_1$ in $D_{k'}$ (or to w_T),

      i) If C_k is a cycle, remove the edge (v_1, v_2) from M_k if it is present. It may be absent if a token was at w^k_1, in that case note that there is a hole at v^k_1.

      ii) If C_k is a cycle, or is odd with the first edge in I or is even with the first edge in F, add the edge (v_{2t_k}, v_{2t_k+1}) to M_k. Note that from the previous step, there is a hole at v_{2t_k+1}.

      iii) If $C_{k'}$ is a cycle or a path with its first edge in I, remove the edge (v_1, v_2) from $M_{k'}$ if it is present.

   c) If the token moves backward from w_{j+1} to w_j in D_k,

i) Remove the edge (v_{2j}, v_{2j+1}) from M_k.

ii) Add (v_{2j+1}, v_{2j+2}) if possible (if there is a hole at v_{2j+2}) to M_k.

d) If the token moves backward from $w_1^{k'}$ in $D_{k'}$ (or from w_T) to $w_{t_k}^k$ in D_k (or to w_0),

i) If C_k is a cycle or odd with its first edge in I or even with its first edge in F, remove (v_{2t_k}, v_{2t_k+1}) from M_k.

ii) If $C_{k'}$ is either a cycle or a path with its first edge in I, add (v_1, v_2) to $M_{k'}$ if possible.

Lemma 4.1 *At each step of the unwinding procedure, the graph* $M = (\bigcup_{k=1}^{z} M_k) \cup (E \setminus (I \oplus F))$ *is a matching.*

Proof: We claim that when a token is at w_j^k, there must be a hole at v_{2j}^k, except possibly in the special case that in the i-th round, B_{2i-1} is at the position of A_{2i}. This implies the lemma since:

a) In the special case that B_{2i-1} is at the position of A_{2i}, B_{2i-1} can only move backwards either from w_{j+1}^k to w_j^k or from $w_1^{k'}$ to $w_{t_k}^k$, and in both these cases, an edge is added only if possible.

b) In every other case, when an edge is added, it is in fact possible.

To see that there is a hole at v_{2j} if a token is at w_j, except in the special case, we go by induction on the number of steps of the token in a round. This is true in the first round after we place the tokens A_2, B_2. In the special case that B_{2i-1} is at the position of A_{2i}, B_{2i-1} can only move backwards either from w_{j+1}^k to w_j^k or from $w_1^{k'}$ to $w_{t_k}^k$, so that there is a hole at v_{2j}^k or $v_{2t_k}^k$ respectively. In every other case, if the token moves to w_j^k, a hole appears at v_j^k. (Since it is true at the end of the round, its true at the start of the next one.) ∎

This lemma and the fact that as the tokens move ahead, edges of I are removed from the M_k and edges of F added, imply that at the end of all the rounds, the above sequence of steps ends with $M = F$.

Let the path M_k be *open* if it matches neither with I nor F. Call it *closed* if it matches with either I or F.

Lemma 4.2 *At each step of the procedure, there are at most 3 distinct paths M_k open with at most 8 holes.*

Proof: Before the start of the first round, all the paths M_k are closed, and match with I. At the start of the first round, upto 2 can be opened, M_1 and say, M_k. Suppose $M_{k'}$ is the last open path for B_2 at the end of the first round.

- If $M_k \neq M_1$ then at the end of the first round, $M_1, M_2, \cdots, M_{k-1}$ are closed (and match with F), and these are never reopened in subsequent rounds.
  - If $M_{k'} \neq M_k$, then $M_k, \cdots, M_{k'-1}$ are closed (and match with F) at the end of the first round and never reopened in subsequent rounds, and $M_{k'}$ is open.
  - If $M_{k'} = M_k$, then M_k is open.
- Suppose $M_k = M_1$, then at the end of the first round,
  - If $M_{k'} \neq M_k (= M_1)$, then $M_1, M_2, \cdots, M_{k'-1}$ are closed (and match with F), and $M_{k'}$ is open with 2 holes at the end of the first round.
  - If however $M_1 = M_k = M_{k'}$, then only M_1 is open, with 2 holes at the end of the first round.

Note that in each case, only $M_{k'}$ is open, with at most 2 holes, and all preceding paths or cycles (if there are any) are closed and never reopened (and match with F). Moreover, the holes are at $v_1^{k'}$ and $v_{2j}^{k'}$ if B_2 is at $w_j^{k'}$. At any time during the first round, there can be at most 3 distinct paths or cycles open: M_{k_1}, M_k, M_{k_2}, for some $1 \leq k_1 < k < k_2 \leq k'$. Moreover, during the unwinding each of M_{k_1}, M_{k_2} each have at most 2 holes at any time and M_k may have upto 4 holes, (though in this case these are the only holes). In every subsequent round, at most one new path is opened. Assume by induction that for every subsequent round, there are at most 2 distinct paths open at the beginning of the round (after A_{2i}, B_{2i} have been placed at their new positions). Again, during the unwinding in this round, there may be at most 3 distinct paths open at any time, where the first and the last have at most 2 holes each, and the middle one may have 4. At the end of the round, there is at most one path open, with at most 2 holes. ∎

4.0.3 Size and Number of Red Edges in Intermediate Matchings

At every step, the sum of the values f at the positions of the tokens is in the interval $[I_R - 1, F_R + 1]$. However, this is not in exact correspondence with the number of Red edges in the intermediate matchings along the path. The discrepancy arises because there are moves when the token B_{2i} is placed at a vertex in the beginning of a round, so that the sum does not change, but an edge of the current matching is removed, which could cause the number of Red edges to decrease. Note that this discrepancy is canceled once the path which was opened by B_{2i} is closed. Hence, at any time, the number of Red edges can be off from the sum by at most 4 (the number of pairs of holes), so that in any intermediate matching on the path M, $I_R - 5 \leq M_R \leq F_R + 1$.

Note that we do not require any restriction on the canonical order of paths and cycles to ensure the condition on the number of Red edges. In order to bound the congestion, however, we also need to ensure that $|M|$ is bounded. Recall that we imposed the following condition on the order of cycles and paths in $I \oplus F$. Suppose that $|I| \leq |F|$ (the other case can be defined analogously). We order first all the alternating cycles and alternating paths with an even number of edges. Since $|I| \leq |F|$ there are exactly $|F| - |I|$ more odd length alternating paths with an extra edge of F than paths with an extra edge of I. Next in the order are (canonically defined) $|F| - |I|$ odd length alternating paths each with an extra edge of F. Lastly, we order the remaining odd length alternating paths so that they alternately contain an extra edge of I or an extra edge of F. This ordering ensures that if the canonical paths are now defined as above, $|I| - 5 \leq |M| \leq |F| + 1$.

5 Bounding the Congestion

Recall that each possible move from M to M' is proposed with probability $1/2|E|$, and performed with probability $\min\left(1, \frac{\pi(M')}{\pi(M)}\right)$. Hence if $P(M, M') > 0$, we have $\pi(M)P(M, M') \geq \frac{1}{2n^2} \min(\pi(M), \pi(M'))$. The length of the path $\gamma_{I,F}$ is bounded by $O(n^2)$ for every pair I, F, by the construction of the paths from Lemma 3.1. Hence,

$$\rho(\Gamma) \leq O(n^4) \max_t \left\{ \frac{1}{\min(\pi(M), \pi(M'))} \sum_{(I,F) \in cp(t)} \pi(I)\pi(F) \right\}. \tag{7}$$

To bound the congestion, we define an injective mapping $E_t : cp(t) \to \Omega$, for every transition t from M to M'. The mapping must be injective in the sense that given $E_t(I, F)$ for any pair $(I, F) \in cp(t)$, and the transition $M \cup M'$, the matchings I and F can be uniquely reconstructed from them. Because there are only 3 paths open at any time, we need to keep track of only constantly many extra edges to define the mapping. We want to keep track of the pairs holes not involved in the current transition. If the first of the two holes is at v_{2j+1}, (an edge was removed to create it), keep track of the edge (v_{2j}, v_{2j+1}) of F just before it on the path, if there is such an edge. If the second hole moved along by sliding to some point v_{2j}, keep track of the edge (v_{2j}, v_{2j+1}) of F just after it (if there is such an edge on the same path). Let H be the set of these edges for the 3 possible pairs of holes not involved in the current transition. Note that $H \subseteq (I \oplus F) \setminus (M \cup M')$ and $|H| \leq 6$. Let

$$E_t(I, F) = \begin{cases} I \oplus F \oplus (M \cup M') - H - (v_1^k, v_2^k) & \text{if } t \text{ is a move switching edges in a cycle } C_k; \\ I \oplus F \oplus (M \cup M') - H & \text{otherwise.} \end{cases}$$

It can be verified that $E_t(I,F)$ is in fact a matching. To see that it is injective, note that, it is easy to reconstruct $I \oplus F$ given $E_t(I,F)$, H and M, M'. Since $|H| \leq 6$, it can be encoded with only a polynomial amount of extra information. To separate the edges of $I \oplus F$ into I and F we use the edges incident with the holes not involved in the transition, and the transition itself. To bound the congestion, we have the following lemma.

Lemma 5.1 *For any $(I,F) \in cp_t$, where t is the transition $M \to M'$,*

$$\pi(I)\pi(F) \leq \frac{1}{\lambda^{10}\mu^{10}}\pi(M)\pi(E_t(I,F)).$$

Proof: Let $S_R = |S \cap R|$ for any $S \subseteq E$. The encoding $E_t(I,F)$ is roughly the complement of M in the union of I and F (viewed as a multiset). By the definition of $E_t(I,F)$, and the fact that $|H| \leq 6$, and the definition of the canonical paths it can be verified that the number of RED edges in the intermediate matching M and the encoding E are roughly bounded by the number of RED edges in the matchings I and F. More formally,

1. $I_R - 5 \leq M_R \leq F_R + 1$;
2. $I_R + F_R - 7 \leq E_R + M_R \leq I_R + F_R$;
3. $I_R - 8 \leq E_R \leq F_R + 5$.

The last set of inequalities follows from the first two. Similarly, we also have

1. $|I| - 5 \leq |M| \leq |F| + 1$;
2. $|I| + |F| - 7 \leq |E| + |M| \leq |I| + |F|$;
3. $|I| - 8 \leq |E| \leq |F| + 5$.

in the case that $|I| \leq |F|$, and the inequalities still hold with I and F interchanged if $|I| \geq |F|$. We first claim that for any k, $|k - I_R| + |k - F_R| \geq |k - M_R| + |k - E_R| - 10$. To see this, we have the following cases, and we use the first 3 inequalities above.

a) If $k \geq \max\{M_R, E_R\}$,

$$\begin{aligned} |k - M_R| + |k - E_R| &= k - M_R \ + \ k - E_R \\ &\leq k - I_R \ + \ k - F_R \ - \ 7 \\ &\leq |k - I_R| \ + \ |k - F_R| \ - \ 7. \end{aligned}$$

b) If $k \leq \min\{M_R, E_R\}$,

$$\begin{aligned} |k - M_R| + |k - E_R| &= M_R - k \ + \ E_R - k \\ &\leq I_R - k \ + \ F_R - k \\ &\leq |k - I_R| \ + \ |k - F_R|. \end{aligned}$$

c) If $M_R < k < E_R$,

$$\begin{aligned} |k - M_R| + |k - E_R| &= k - M_R \ + \ E_R - k \\ &\leq k - (I_R - 5) \ + \ (F_R + 5) - k \\ &\leq |k - I_R| \ + \ |k - F_R| \ + \ 10. \end{aligned}$$

d) If $E_R < k < M_R$,

$$\begin{aligned} |k - M_R| + |k - E_R| &= M_R - k \ + \ k - E_R \\ &\leq (F_R + 1) - k \ + \ k - (I_R - 8) \\ &\leq |k - I_R| \ + \ |k - F_R| \ + \ 9. \end{aligned}$$

By the same computations, for any ℓ, $|\ell - |I|| + |\ell - |F|| \geq |\ell - |M|| + |\ell - |E|| - 10$, and these two inequalities imply the Lemma. ∎

By inequality (7) and the fact that H can be encoded by a polynomial amount of information, the lemma implies that for some polynomial p,

Theorem 5.2 *For $0 < \lambda, \mu \leq 1$ the congestion $\rho(\Gamma) \leq O\left(\frac{p(n)}{\lambda^{10}\mu^{10}}\right)$.*

This implies Theorem 2.1 as a corollary.

5.1 Canonical Paths for the Markov Chain $\mathcal{T}_2$

Recall that the state space of $\mathcal{T}_2$ is the set $\cup_{i=1}^{4} \mathcal{N}^i \cup \mathcal{P}$. We now define the canonical paths.

Let $I \in \mathcal{N}^i$ for $1 \leq i \leq 4$. Define a matching $\widehat{I}$ in $\mathcal{N}^{i-1}$ as follows. Since the graph G is dense, a simple counting argument shows that for a pair of unmatched vertices u, v (in different bipartitions if the graph is bipartite) either there is an alternating path of length at most 3 from u to v or there is some vertex v' in the same bipartition as v so that the edge (u, v') is in the graph. Let $\widehat{I}$ be the matching obtained from I by augmenting along this path if it exists. If the path does not exist, $\widehat{I}$ is obtained by augmenting I by matching u to v'. Clearly each $\widehat{I} \in \mathcal{N}^{i-1}$ has at most a polynomial number of preimages I. We note that this construction is exactly the one used by Jerrum and Sinclair [7] to define canonical paths between near-perfect matchings in the monochromatic setting. In our case, the fact that the augmentation involves only a constant number of edges gives a bound on the change in the number of Red edges, which is required to bound the congestion.

If $(I, F) \in \mathcal{P} \times \mathcal{P}$, then the path is defined as it is for $\mathcal{T}_1$. If $I, F \in \mathcal{N}^i \times \mathcal{P}$, start the path at I and use the construction above repeatedly, until we reach a matching in $\mathcal{P}$. We then use the paths in the first case to get to F. We can bound the congestion due to pairs of perfect matchings as in Theorem 5.2 and the congestion due to the matchings with upto 8 unmatched vertices is only an additional polynomial factor by the construction above. A similar argument is given in [7], and we use the additional fact that the augmentations from I to $\widehat{I}$ involve only constantly many edges. Thus the number of Red edges in matchings along the canonical path remains bounded as desired. As a corollary we obtain Theorem 2.2.

6 Bichromatic Matchings on the Torus

We now demonstrate how Theorems 2.1 and 2.2 can be applied when the underlying graph is a lattice region and we are interested in configurations with a fixed number of vertical or horizontal edges. Let G_{m_1,m_2} be the torus $\mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2}$ with horizontal edges colored Red and vertical edges Blue, where m_1 and m_2 are even. It is known that the total number of near perfect matchings is polynomially related to the number of perfect matchings ignoring colors [12]. The proof used an injection from $\mathcal{N}(u,v)^2$ to $\mathcal{P}^2$, where here $\mathcal{N}(u,v)$ denotes the set of matchings whose unmatched vertices are u and v.

We first generalize this result to relate near-perfect matchings restricted to k Red edges and perfect matchings on close to k Red edges. Our goal will be to show that the number of matchings with k red edges does not vary much as we change the size of the matching. We will show this by defining maps from one set of matchings to the other that are invertible with a small amount of additional information. This will allow us to define an fpras for counting the number of (k,ℓ)-matchings of the torus. Let $\mathcal{N}_k^i$ be the set of $(k, m_1m_2/2 - i)$-matchings of G_{m_1,m_2}, recall that this means matchings with k Red edges and $2i$ unmatched vertices. Let $\mathcal{P}_k = \mathcal{N}_k^0$ be the set of perfect matchings with k Red edges. Let $\mathcal{N}_k(u,v)$ be the set of $(k, m_1m_2/2 - 1)$ matchings with holes at u and v.

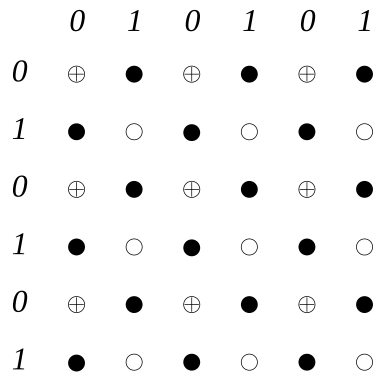

Figure 4: The sublattice V_{00} ($\oplus$)

Since m_1, m_2 are even, G_{m_1,m_2} is bipartite. Let V_0 (white vertices) and V_1 (black vertices) be the even and odd sublattices of G_{m_1,m_2}. Further refining these sets, let V_{00} be the set of vertices both of whose co-ordinates are even: the sets V_{01}, V_{10} and V_{11} are defined analogously (Figure 4) . Note that if u, v are the unmatched vertices of any near-perfect matching of G_{m_1,m_2}, then one of them is white while the other is black. Also, if m_1, m_2 are even, the number of RED edges (and hence the number of BLUE edges) in any perfect matching of G_{m_1,m_2} is even.

Theorem 6.1 *Let $m_1, m_2 \in \mathbb{Z}$ be even and let $N = m_1 m_2/2$. For $0 \le i \le N-1$ and $1 \le k \le N-1-i$, there is a map $f_i : \mathcal{N}_k^{i+1} \to \mathcal{N}_k^i \cup \mathcal{N}_{k+1}^i$ such that no matching of $\mathcal{N}_k^i \cup \mathcal{N}_{k+1}^i$ is the image of more than $O(N^3)$ matchings in $\mathcal{N}_k^{i+1}$.*

The idea of the proof is to construct augmenting paths from one unmatched vertex to another to define a matching with one additional edge. We need to do this in such a way that the path can be recovered after the augmentation is done with only a small amount of additional information. In order to preserve the number of horizontal edges, an augmenting path will consist of horizontal and vertical segments of even length except at the beginning or end.

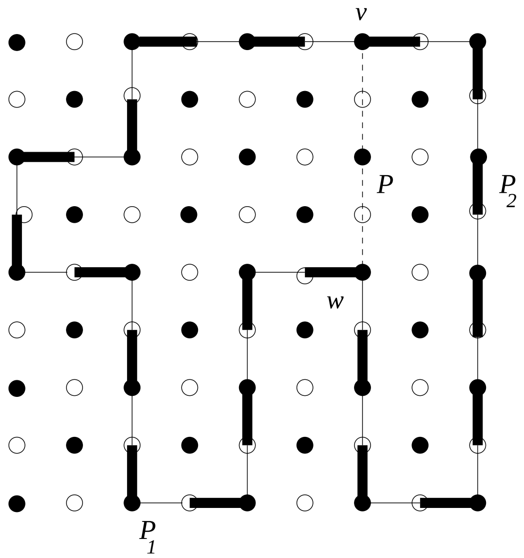

Figure 5: Length of the top segment ≥ 4.

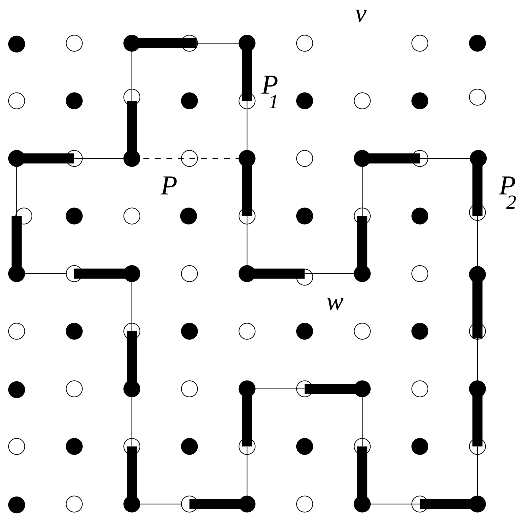

Figure 6: Length of the top segment ≤ 2.

We will first prove a useful lemma, for which we need some notation. Let C_1 be a contractible cycle on the 2-d torus G_{m_1,m_2} such that each vertical and horizontal segment is of even length. Let every black vertex of C_1 be on the sublattice V_{01}. Then the white vertices on the horizontal segments are in V_{00} and those on the vertical segments are in V_{11}. In topology, cutting along a contractible cycle on the torus results in two pieces, one of which can be embedded into a finite region of the 2-d lattice, which is the interior. Let the set of vertices in the interior of C_1, not including the vertices of C_1 be called $Int(C_1)$.

Lemma 6.2 *The number of vertices in $Int(C_1)$ is odd, and the number of black vertices in the interior exceeds the number of white vertices by 1.*

Proof: The proof is by induction on the number of vertices in $Int(C_1)$. The lemma is clearly true when $Int(C_1)$ consists of a single vertex.

Any contractible cycle can be embedded into the lattice $\mathbb{Z}^2$ as a contractible cycle, fix such an embedding. Note that the embedding preserves the number of vertices in the interior. Consider any horizontal segment that maximizes the y-coordinate in this embedding. First suppose that its length is at least 4 (see Figure 5). Let v be a black vertex on this horizontal segment so that it is not a "corner". Construct an even length vertical path P starting at v and ending at w, where w is the first vertex of C_1 the path hits. Note that w must be a black vertex in V_{01} by the definition of C_1. Hence $P \cap Int(C_1)$ is odd. Then P divides the cycle into two paths P_1, P_2, one going from v to w along C_1 in one direction, and the other going from w to v along C_1 in the same direction. $P_1 \cup P$ and $P_2 \cup P$ are two contractible cycles with strictly fewer vertices in their interior, since $P \cap Int(C_1) > 1$. Moreover, both have every vertical and horizontal segment even, and their black and white vertices lie on the same sublattice as the vertices of C_1. Finally since $P \cap Int(C_1)$ is odd, $|Int(C_1|)$ is odd as well. The second part of the lemma follows by induction since both $Int(P \cap P_1)$ and $Int(P \cap P_2)$ have one extra black vertex while $P \cap Int(C_1)$ has an extra white vertex.

Now suppose that every topmost horizontal segment of C_1 is of length two (see Figure 6). In this case, we go by induction, by dividing the cycle into two smaller cycles using the path P as shown. Note that P has non-empty intersection with $Int(C_1)$ since $|Int(C_1)| > 1$. ∎

We now prove Theorem 6.1. The key is defining an augmenting path to preserve the number of horizontal edges.

Proof of Theorem 6.1: We first prove the theorem for $i = 0$ where we change near-perfect matchings to perfect matchings. Let $N \in \mathcal{N}_k(u, v)$ and assume wlog that $u = (u_1, u_2)$ is in V_{00}. Define an alternating path L_1^u in N as follows: start at $z_0 = u$, and follow the unmatched RED edge to the vertex $z_1 = (u_1, u_2+1)$. Now, iteratively, if at an odd vertex z_{2i-1}, follow the unique matched edge to z_{2i} (see figure 7). From each even vertex z_{2i} along the path, take the unmatched edge in the *same direction* as the edge (z_{2i-1}, z_{2i}). Continue in this way so each segment (in the horizontal or vertical direction) of the path after the first step has even length. The path terminates when it reaches v or revisits a vertex, thus forming a cycle. The vertex set is finite, so one of these will occur. Now define a second alternating path L_2^u similarly, except, start with the edge from u to $(u_1, u_2 - 1)$. Note that every black vertex on these paths is in V_{01}, while the white vertices along the horizontal segments are in V_{00}, and those on the vertical segments are in V_{11}. Finally, define the paths K_1^u, K_2^u similarly, so that the first edges are to the vertices $(u_1 \pm 1, u_2)$. In this case, the black vertices on the path are in V_{10}, the white vertices on the vertical segments are in V_{00} and those on the horizontal segments are in V_{11}.

First, suppose that $v = (v_1, v_2) \in V_{01}$. We use these four paths to define an alternating path from u to v where the number of Red unmatched edges on the path is exactly one more than the number of Red matched edges. Switching matched and unmatched edges along this path gives a perfect matching in $\mathcal{P}_{k+1}$. Given a perfect matching obtained in this way, we can recover the near perfect matching with only a small amount of information. We now show how to define the alternating path from u to v by case-analysis.

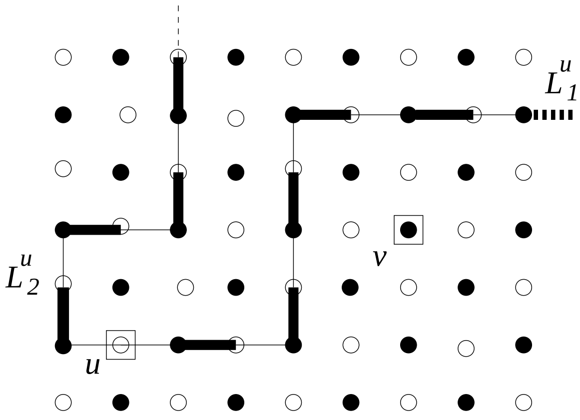

Figure 7: Alternating paths L_I^u, L_2^u

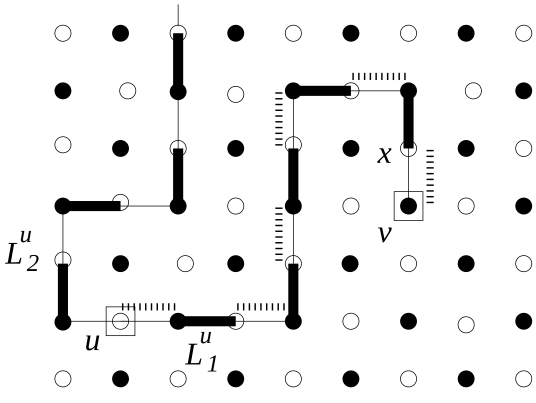

Figure 8: L_1^u meets v

1. If one of the paths L_1^u or L_2^u reaches the vertex v before it cycles (Figure 8), then this is the alternating path. Say the path L_1^u reaches v. By construction, the number of unmatched Red edges along L_1^u is exactly one more than the number of matched Red edges, hence inverting along the path gives $P \in \mathcal{P}_{k+1}$. To invert the map, given $P \in \mathcal{P}_{k+1}$, start at v, if v is matched by a Blue (Red) edge to the vertex x, the next unmatched edge along the path is taken to be the other Blue (Red) edge incident with x. Continue in this

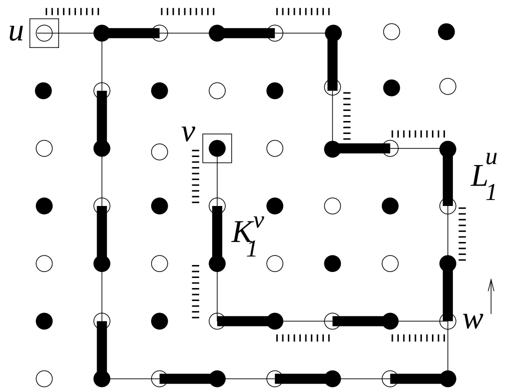

Figure 9: The path K_1^v meets C_1

way until u is reached. We only need a couple of bits information to encode that we are in this case.

2. If both paths L_1^u, L_2^u cycle without reaching v, we consider sub-cases based on whether or not these cycles are contractible.

(a) Suppose at least one of the paths, say L_1^u , ends with a contractible cycle C_1 on the torus. By Lemma 6.2, the interior of C_1 must contain an odd number of unmatched vertices and the unmatched vertex in the interior cannot be white: in particular, it cannot be u itself. So, v must lie in the interior of C_1. Now consider the path K_1^v (see Figure 9). Since K_1^v cannot cycle in the interior of C_1 or end at an unmatched vertex, it must intersect C_1. By construction, the white vertices of K_1^v are in V_{11}, hence the path meets the cycle on a vertical segment, say at the vertex w. The alternating path from u to v is defined by taking the subpath of L_1^u from u to w, and then the subpath of K_1^v from v to w. The number of unmatched Red edges in the alternating path is indeed one more than the number of matched Red edges, and so inverting along this path gives $P \in \mathcal{P}_{k+1}$ as required. Moreover, given $u, v, P$and the edge incident to w in the original matching, the alternating path can be uniquely reconstructed.

(b) Suppose instead that both paths L_1^u, L_2^u end in non-contractible cycles on the surface of the torus. We consider two possibilities in this case:

i) The cycles C_1 and C_2 are disjoint. This implies that the paths L_1^u and L_2^u must be disjoint except at u. When a torus is "cut" along an non-contractible simple cycle, it topologically forms a cylinder. If we cut along the disjoint cycles C_1 and C_2, we are left with 2 cylinders, one of which contains u and the paths L_1^u and L_2^u (including the vertices of C_1 and C_2). It can be easily verified that both cylinders have an even number of vertices. Since the union of the two paths L_1^u, L_2^u is odd, the cylinder containing the paths has an odd number of vertices not including $L_1^u \cup L_2^u$, and hence contains the vertex v. Now, K_1^v must intersect one of the paths L_1^u or L_2^u since it cannot cycle on the cylinder without doing so. K_1^v hits the path from u on a vertical segment at a white vertex in V_{11}. Given a matching in $\mathcal{P}_{k+1}, u, v$ and the vertex at which the paths from u and v meet, we can invert the map as described before.

ii) The cycles C_1, C_2 are not disjoint. In this case it can be verified that there exists a contractible cycle on the torus starting at u along L_1^u, and ending at u along L_2^u (some edges may be used twice, once in each direction). As before, the interior must contain an odd number of vertices that must be matched with each other, and hence it must contain the vertex v. Since the cycle containing v is contractible, the path K_1^v must hit one of L_1^u, L_2^u. This is again sufficient to define the augmenting path.

In the case that $v \in V_{10}$, the same arguments can be made, except that we consider the paths K_1^u, K_2^u initially and then L_1^v, L_2^v instead of L_1^u, L_2^u first and K_1^v, K_2^v later. The difference is that the alternating paths constructed have one unmatched Blue edge more than the number of matched Blue edges along the path, so inverting edges with non-edges along the alternating path from u to v gives a matching in $\mathcal{P}_k$. This completes the proof when $i = 0$.

We now use similar arguments to complete the proof in the general case where $i > 0$. Suppose that $N \in \mathcal{N}_k^{i+1}$. Let u be the lexicographically first unmatched white vertex of N, and assume that $u \in V_{00}$. If one of L_1^u, L_2^u meets a black unmatched vertex v, then switching edges along the path from u to v gives a matching in $\mathcal{N}_{k+1}^i$. If not, then both L_1^u, L_2^u cycle.

Suppose L_1^u ends in a contractible cycle C_1. The interior of C_1 contains an odd number of vertices, including the vertices possibly on a segment of L_1^u starting at u. Hence, the interior contains an odd number of unmatched vertices. Since black vertices outnumber white vertices by one, the number of black unmatched vertices outnumber white unmatched vertices by one. In particular, the interior contains at least one black unmatched vertex, that we call v.

Now consider the paths K_1^v, K_2^v. If either one reaches a white unmatched vertex in the interior (including u), then switching edges along that path gives a matching in $\mathcal{N}_k^i$. Otherwise, if either path hits L_1^u, say at a vertex w, then we can switch edges along L_1^u from u to w, and then along K_1^v from v to w, to obtain a matching either in $\mathcal{N}_{k+1}^i$ or $\mathcal{N}_k^i$ depending on whether v is in V_{01} or V_{10}. If the paths K_1^u, K_2^v do not hit a white unmatched vertex or L_1^u, they must cycle in a contractible cycle in the interior. Consider one of the paths, say K_1^v. Repeat the same argument as before, except now we consider white vertices u' in the interior of the cycle, and consider the paths $L_1^{u'}, L_2^{u'}$. Depending on whether u' is in V_{00} or V_{11} and the sublattice of v, alternating along the paths as before gives a matching either in $\mathcal{N}_k^i$ or $\mathcal{N}_{k+1}^i$. We can repeat this argument until we obtain an alternating path between a black and a white vertex, or, the interior of some cycle created by a vertex contains only one unmatched vertex. Since the single unmatched vertex cannot be the same as the vertex from which the cycle was created, this case can be solved in the same manner as previously when $i = 0$.

The remaining case, when L_1^u, L_2^u end in non-contractible cycles, is similar to Case ii) above. In all cases, we can define an appropriate augmenting path with the needed number of RED edges and the Theorem follows. ∎

The above theorem allows us to define canonical paths from nearly perfect matchings on the torus to perfect matchings so that the number of RED edges does not change by more than a constant for each pair of holes that are matched. This gives the following corollary, which shows that we can define an fpras to estimate the partition function over perfect matchings with exactly k RED edges.

Corollary 6.3 *Let m_1, m_2 be even, $N = m_1 m_2/2$. There is an algorithm to estimate the partition function $Z_\lambda^k(\mathcal{P})$ given in Equation (2) for every $\lambda \leq 1$ and k to within $(1 \pm \varepsilon)$ w.p. $\geq 1 - \delta$ in time polynomial in $N, \lambda, 1/\varepsilon$ and $\log(1/\delta)$.*

In addition, we can use similar arguments to relate the number of perfect matchings with k or $k+2$ RED edges.

Theorem 6.4 *Let m, n be even, $N = mn/2$. For every $0 \leq k \leq N-2$ even, $|\mathcal{P}_{k+2}|/p(N) \leq |\mathcal{P}_k| \leq p(N)|\mathcal{P}_{k+2}|$, where $p(n) = O(n^3)$.*

Proof: It suffices to show the upper bound for all k since the lower bound follows by inverting the colors.

We construct a map from $\mathcal{P}_k$ to $\mathcal{P}_{k+2}$ as follows. Let $P \in \mathcal{P}_k$. Delete any vertical edge (u, v). Since $k \leq N-2$, there must be such an edge. Consider the paths L_1^u, L_2^u in $P \setminus (u, v)$. By parity, neither can reach v, and hence they must cycle on the surface of the torus. Since u is adjacent to v on the torus, neither path can end in a contractible cycle containing v in the interior. Hence, both L_1^u and L_2^u end in non-contractible cycles. By the arguments of Case 2 of the previous theorem, the path L_1^v must hit one of the paths L_1^u or L_2^u at a white vertex $w \in V_{11}$, i.e., on a vertical segment. Then, switching along the alternating path from u to v through w as before, we gain two RED edges, giving a matching in $\mathcal{P}_{k+2}$. The mapping is invertible given the vertices u, v and w, hence $|\mathcal{P}_k| \leq O(N^3)|\mathcal{P}_{k+2}|$. ∎

Using this Theorem and the estimator given by Corollary 6.3, we obtain an estimator for the set of perfect matchings of the torus with exactly k RED edges.

Theorem 6.5 *There is an algorithm to estimate $|\mathcal{P}_k|$ to within $1 \pm \varepsilon$ for every $0 < \varepsilon < 1$ with probability $\geq 1 - \delta$ in time polynomial in $N, 1/\varepsilon$ and $\log(1/\delta)$.*

The proof follows from standard arguments, along the lines of those in [16], see Proposition 3.8.

These results can be generalized to approximating the size of the set of (k,ℓ)-matchings for any ℓ. By Theorem 1.1, we can approximate the partition function $Z^{k,\ell}_{\lambda,\mu}(\mathcal{M})$ given in Equation (1) for every $\lambda,\mu \le 1$ and $0 \le k \le \ell \le n$. This estimator, together with the relations among sets of restricted matchings of arbitrary size (stated below) and the theorem of Kenyon, Randall and Sinclair [12] showing that the sizes of the sets $\mathcal{N}^i$ and $\mathcal{N}^{i+1}$ are polynomially related, gives an approximate counter for sets of restricted matchings of any size.

Theorem 6.6 *Let m_1, m_2 be even, $N = m_1 m_2/2$. For every $1 \le i \le N-1$, and $0 \le k \le N-i-1$, then for some polynomial p, $|\mathcal{N}^i_{k+1}|/p(N) \le |\mathcal{N}^i_k| \le p(N)|\mathcal{N}^i_{k+1}|$.*

Proof: The idea of the proof is to construct an injective map from $\mathcal{N}^i_k$ to $\mathcal{N}^i_{k+1}$ which is invertible given a small (polynomial) amount of information. Note that this gives the lower bound in the theorem. It also implies the upper bound since we can prove the same relations for Blue edges.

Let $N \in \mathcal{N}^i_k$. Since $k < N-i$, N contains a Blue edge (x,y). Let u be an unmatched white vertex of N in the V_{00} sublattice. Consider the paths L^u_1, L^u_2. If one of them reaches a black vertex v, the switching the edges and non-edges along the path gives a matching in $\mathcal{N}^{i-1}_{k+1}$. If we delete the edge (x,y), we obtain a matching in $\mathcal{N}^i_{k+1}$ as required.

First suppose that both L^u_1, L^2_u cycle. In all three possibilities for the cycles defined in the proof of Theorem 6.1, there must be a contractible region of the torus R_u. This region consists of an odd set of vertices bounded by edges from L^u_1 and L^2_u, with one extra black vertex, that contain some matched edges. Hence, R_u must contain an odd number of unmatched vertices, and the number of black unmatched vertices exceeds the number of white unmatched vertices by one, as before. In particular, there is at least one unmatched black vertex, v in R_u.

Consider the paths L^v_1, L^v_2. If one of them reaches a white unmatched vertex in R_u, switching along this path gives a matching in $\mathcal{N}^{i-1}_{k+1}$. If possible, we delete (x,y) to obtain a matching in $\mathcal{N}^i_{k+1}$. It may not be possible if (x,y) was removed while switching along the path. However, one can check in this case (x,y) was switched with some other Blue edge (x',y'), and we remove this edge instead.

A second possibility is that one of the paths, say L^v_1 hits L^u_1 or L^u_2 at some vertex w. If $v \in V_{10}$, switching along the alternating path from u to w and then to v along L^v_1 gives a matching in $\mathcal{N}^{i-1}_{k+2}$. We can delete the Red edge adjacent to u to obtain a matching in $\mathcal{N}^i_{k+1}$. If $v \in V_{01}$, switching along the alternating path from u to w and then to v along L^v_1 gives a matching in $\mathcal{N}^{i-1}_{k+1}$. Again, we remove the Blue edge (x,y) or (x',y') to obtain a matching in $\mathcal{N}^i_{k+1}$.

The third possibility is that the path L^v_1 cycles in a contractible cycle C_v within R_u. By the symmetry between black and white vertices, the interior of this cycle R_v contains one more unmatched white vertex than black unmatched vertices, hence the total number of unmatched vertices in R_v is strictly fewer than the number in R_u. Hence, we repeat the arguments above, interchanging the role of black and white vertices, until the interior of some contractible region has only one unmatched vertex so that the paths from it cannot cycle. ∎

The following corollary puts together the pieces to verify that we indeed have an fpras for $|\mathcal{N}^i_k|$.

Corollary 6.7 *There is an algorithm to estimate $|\mathcal{N}^i_k|$ to within a factor of $1 \pm \varepsilon$ for every $0 < \varepsilon < 1$ with probability $\ge 1-\delta$ in time polynomial in $N, 1/\varepsilon$ and $\log(1/\delta)$.*

References

[1] N. Bhatnagar, D. Randall, V.V. Vazirani, E. Vigoda. Random Bichromatic Matchings. *Proceedings of the 7th Latin American Theoretical Informatics Symposium*, (2006), pp. 190-201.

[2] R. Burton and R. Pemantle. Local characteristics, entropy and limit theorems for spanning trees and domino tilings via transfer-impedances. *Annals of Probability* **21(3)**(1993), pp. 1329-1371.

[3] M.E. Fisher. Statistical Mechanics of Dimers on a Plane Lattice. *Phys. Rev.* **124** (1961), pp.1664-1672.

[4] R.H. Fowler and G.S. Rushbrooke. Statistical theory of perfect solutions. *Transactions of the Faraday Society* **33** (1937), pp. 1272-1294.

[5] O.J. Heilmann and E.H.Leib. Theory of monomer-dimer systems. *Communications in Mathematical Physics* **25** (1972), pp. 190-232.

[6] M. Jerrum. Counting, Sampling and Integrating: Algorithms and Complexity. Birkhauser, (2003).

[7] M. Jerrum and A. Sinclair. Approximating the Permanent. *SIAM Journal of Computing* **18** (1989), pp. 1149-1178.

[8] M. Jerrum, A. Sinclair, and E. Vigoda. A Polynomial-Time Approximation Algorithm for the permanent of a matrix with non-negative entries. *Journal of the ACM* **51(4)** (2004), pp. 671-697.

[9] M. Jerrum, L. Valiant, and V. Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theoretical Computer Science* **43** (1986), pp.169-188.

[10] A.V. Karzanov. Maximum Matching of given weight in complete and complete bipartite graphs. *Kibernetica: 7-11, English translation in CYBNAW* **23(1)** (1987) pp. 8-13.

[11] P. W. Kasteleyn. The statistics of dimers on a lattice,I. The number of dimer arrangements on a quadratic lattice. *Physica* **27** (1961) pp. 1209-1225.

[12] C. Kenyon, D. Randall, and A. Sinclair. Approximating the number of monomer-dimer coverings of a lattice. *Journal of Statistical Physics* **83** (1996) pp. 637-659.

[13] M. Matamala. *Personal Communication.* (2006).

[14] K. Mulmuley, U.V. Vazirani, and V. V. Vazirani. Matching is as easy as matrix inversion. *Combinatorica* **7(1)** (1987), pp.105-113.

[15] C. Papadimitriou. Polytopes and Complexity. *Progress in Combinatorial Optimization* W.R. Pulleyblank ed., Academic Press Canada, Don Mills, (1984) pp. 295-305.

[16] A. Sinclair. Algorithms for Random Generation and Counting : A Markov Chain Approach *Progress in Theoretical Computer Science.* Birkhauser, (1993)

[17] H. N. V. Temperley. In Combinatorics, Proceedings of the British Combinatorial Conference 1973, London Mathematical Society Lecture Notes Series No.13. (1974), pp. 202-204.

[18] T. Yi, K. G. Murty, C. Spera. Matchings in colored bipartite networks. *Discrete Applied Mathematics* **121(1-3)** (2002), pp. 261-277.